Notebook

March 9, 2025

root/RAG_const/text_dataset_RAG_contruction.ipynb

[1]:
```
import json
import torch
import numpy as np
import faiss
from tqdm import tqdm
from transformers import BertTokenizer, BertModel

# **   BERT**
LOCAL_MODEL_PATH = "/root/autodl-tmp/all-MiniLM-L6-v2"  #

#    tokenizer
tokenizer = BertTokenizer.from_pretrained(LOCAL_MODEL_PATH)
model = BertModel.from_pretrained(LOCAL_MODEL_PATH)

#
def load_text_data(json_file):
    with open(json_file, "r", encoding="utf-8") as f:
        data = json.load(f)
    texts = []
    for item in data:
        if item["type"] == "qa":
            texts.append(f"Q: {item['question']} A: {item['answer']}")
        elif item["type"] == "textbook":
            texts.append(item["text"])
    return texts

#   BERT
def get_embedding(text):
    inputs = tokenizer(text, return_tensors="pt", truncation=True,␣
 ↪padding=True, max_length=512)
    with torch.no_grad():
        outputs = model(**inputs)
    return outputs.last_hidden_state[:, 0, :].squeeze().numpy()  #   [CLS]

#   FAISS
def build_faiss_index(texts):
    print("\n    ...")
```

```
    embeddings = np.array([get_embedding(text) for text in tqdm(texts,␣
 ↪desc="  ")], dtype="float32")

    print("\n  FAISS   ...")
    dimension = embeddings.shape[1]
    index = faiss.IndexFlatL2(dimension)
    index.add(embeddings)
    return index, texts

#   FAISS
def save_retrieval_system(index, texts, index_file, texts_file):
    print("\n  FAISS     ...")
    faiss.write_index(index, index_file)
    with open(texts_file, "w", encoding="utf-8") as f:
        json.dump(texts, f, ensure_ascii=False, indent=4)
    print("         ")

# **  **
def build_and_save_text_retrieval_system(json_file, index_file, texts_file):
    texts = load_text_data(json_file)
    index, texts = build_faiss_index(texts)
    save_retrieval_system(index, texts, index_file, texts_file)

#
text_data_file = "/root/autodl-tmp/updated_data.json"
text_index_file = "/root/autodl-tmp/text_index.faiss"
text_texts_file = "/root/autodl-tmp/text_texts.json"

#
build_and_save_text_retrieval_system(text_data_file, text_index_file,␣
 ↪text_texts_file)
```

```
  …
: 100%|      | 276075/276075 [41:12<00:00, 111.65it/s]

 FAISS   …

 FAISS     …
```

This notebook was converted with convert.ploomber.io